Abstract

An application programming interface (API) protocol is provided for making requests to registered applications regarding applications' dependency information so that a table of dependency information relating to a target object can be recursively generated. When all of the applications' dependencies are captured at the same time for given volume(s) or object(s), the entire volume's or object's program and data dependency information may be maintained for the given time. With access to this dependency information, the computer system advantageously knows not only which files and in which order to freeze or flush the files in connection with a backup, such as a snapshot, or restore of given volume(s) or object(s), but also knows which volume(s) or object(s) can be excluded from the freezing process.

After a request by a service for application dependency information, the computer system can translate or process dependency information, thereby ordering recovery events over a given set of volumes or objects. The set of volumes or objects may be co-located or remotely located relative to one another via a networked environment. In a preferred embodiment, dependency information is generated in connection with an iterative collaboration process between the system and applications, and a common protocol for an application programming interface (API). Thus, applications can communicate their external dependencies to a common software agent having a storage component, thereby maintaining a table, list or other organized information relating to applications' dependencies for a target object for a consistent point in time. This table, list or other organized information advantageously can be used as part of a service such as a backup, restore or recovery process, or any service wherein consistent point in time application dependency information is useful to the service.